ABRS Label Maker GUI

How-to-use Guide

Created by Primoz Ravbar
Edited by Venk Muriki
Simpson Lab
University of California, Santa Barbara

Table of Contents

Welcome

The [Automatic Behavior Recognition System (ABRS)] classifies animal behaviors using movements and their temporal context. The ABRS Label Maker Graphical User Interface (GUI) was designed in Python so that users can generate ethogram labels and spatio-temporal (ST) images via a machine learning model from video footage of the animal.

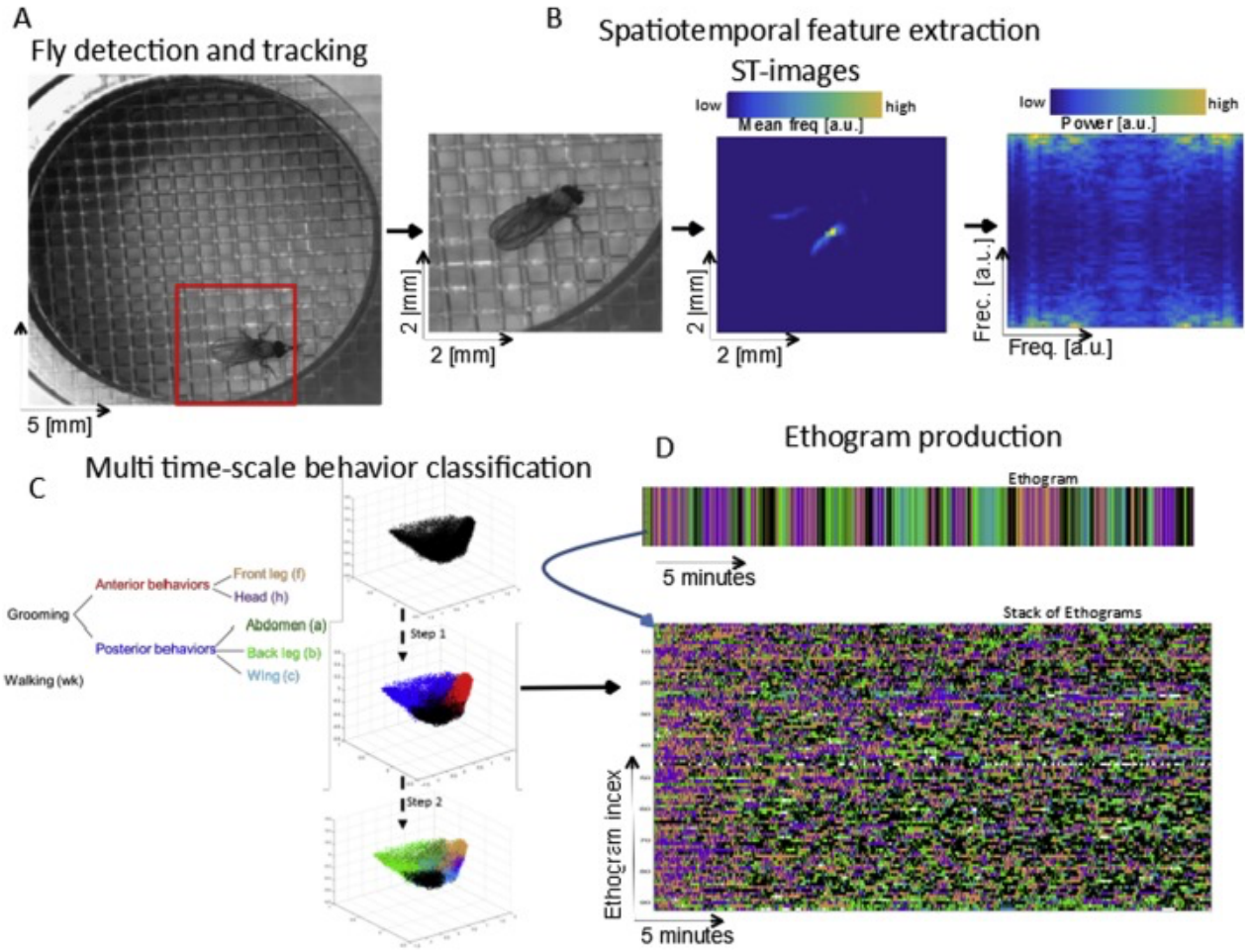


Image of how ABRS works.
Credit: https://www.sciencedirect.com/science/article/pii/S0165027019302092

Setting Up

1. **Install Python**

Here is where you can install python based on if you have Windows or Mac: https://www.python.org/downloads/
Even if you have python, it is recommended that you update it to the latest version.

2. **Install Anaconda/Conda**

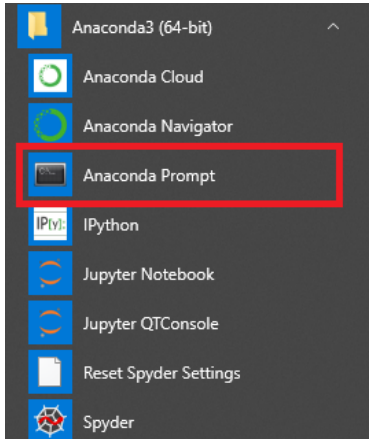
Link: https://conda.io/projects/conda/en/latest/user-guide/install/index.html

→ Click regular installation instructions based on the OS and download Anaconda not Miniconda. Click on the 64-Bit Graphical Installer (435 MB) at the bottom of the page based on Windows or Mac.

Follow those instructions for installations and continue.

3. **Open up command prompt applications**

- Windows: Type "Anaconda Prompt" in start search and click on that app (cmd on Windows will be used later)



- Mac: Open Terminal.

In either Windows or Mac, type `conda -V` exactly and press <Enter>/<Return>. The output should relay the conda version on the system. If not, installation of Anaconda was not successful. If you are on a Mac and get a problem like "conda not found" or "conda not recognized", please watch this video for a solution: https://www.youtube.com/watch?v=IeBCEn-Ybqo&ab_channel=BoostUpStation.

4. Install Packages

Now there are some additional Python/machine learning packages that need to be separately installed that do not already come with Anaconda.

Warning: If at any time in the Terminal/AnacondaPrompt when you type the following package installation commands, and you get a hang on "Solving the environment", try updating conda like this: `conda update --all`. If that does not fix the lag, check out this page of other solutions that could also work: https://stackoverflow.com/questions/52721363/conda-hangs-in-solving-environment-when-installing-stsci-packages

A. To install the **mplcursors** library, in the Terminal (Mac)/AnacondaPrompt (Windows), run the command like so: Type: `conda install -c conda-forge mplcursors` and press <Enter>/<Return>.
B. Similarly, to install **opencv**, run: `conda install -c conda-forge opencv`
C. To install the **tensorflow** library,
    a. for Mac, in the Terminal, run:
    b. for Windows, open up cmd (instead of AnacondaPrompt) and run:

```
pip install --upgrade pip
```

and then run:

```
pip install tensorflow
```

That concludes the set-up process for the ABRS GUI!

JupyterLab

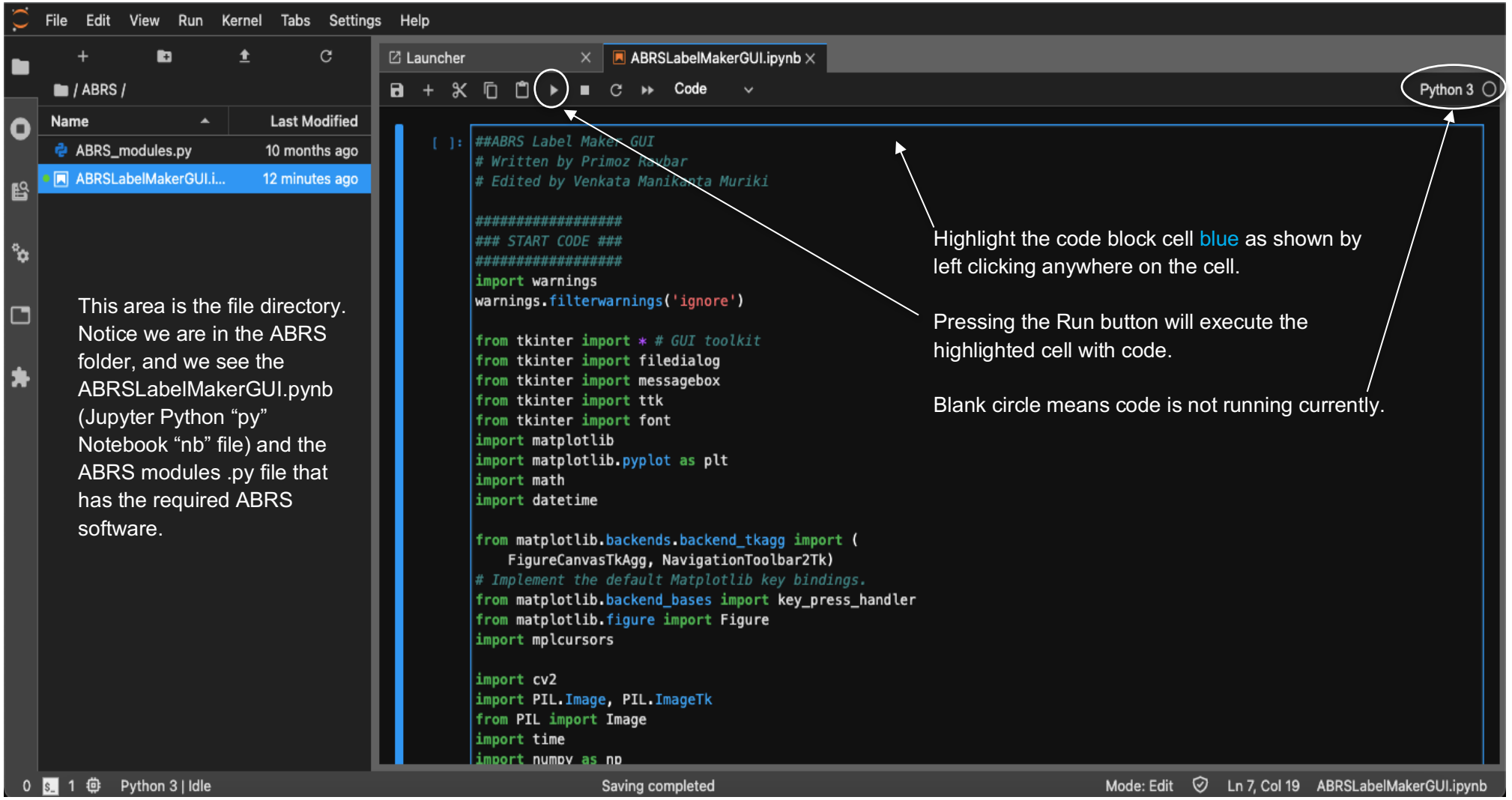
JupyterLab is a professional, web-based interactive development environment for code, notebooks, data, graphics, and more. The GUI should be run on this environment to ensure that all the code executes properly and runs the program thoroughly, since development and testing has been completed on JupyterLab.

To access JupyterLab, run `jupyter lab` on Terminal (Mac) or AnacondaPrompt (Windows). Or you can search for an application called Anaconda Navigator on either operating system and then open JupyterLab from there. A window/tab will open up JupyterLab on your default internet browser.

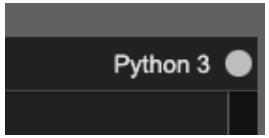
On JupyterLab, on the left-hand side, navigate to the unzipped ABRS file named ABRSLabelMakerGUI.ipynb. A block of code should open (see image below). You are now set to run the GUI!

First run

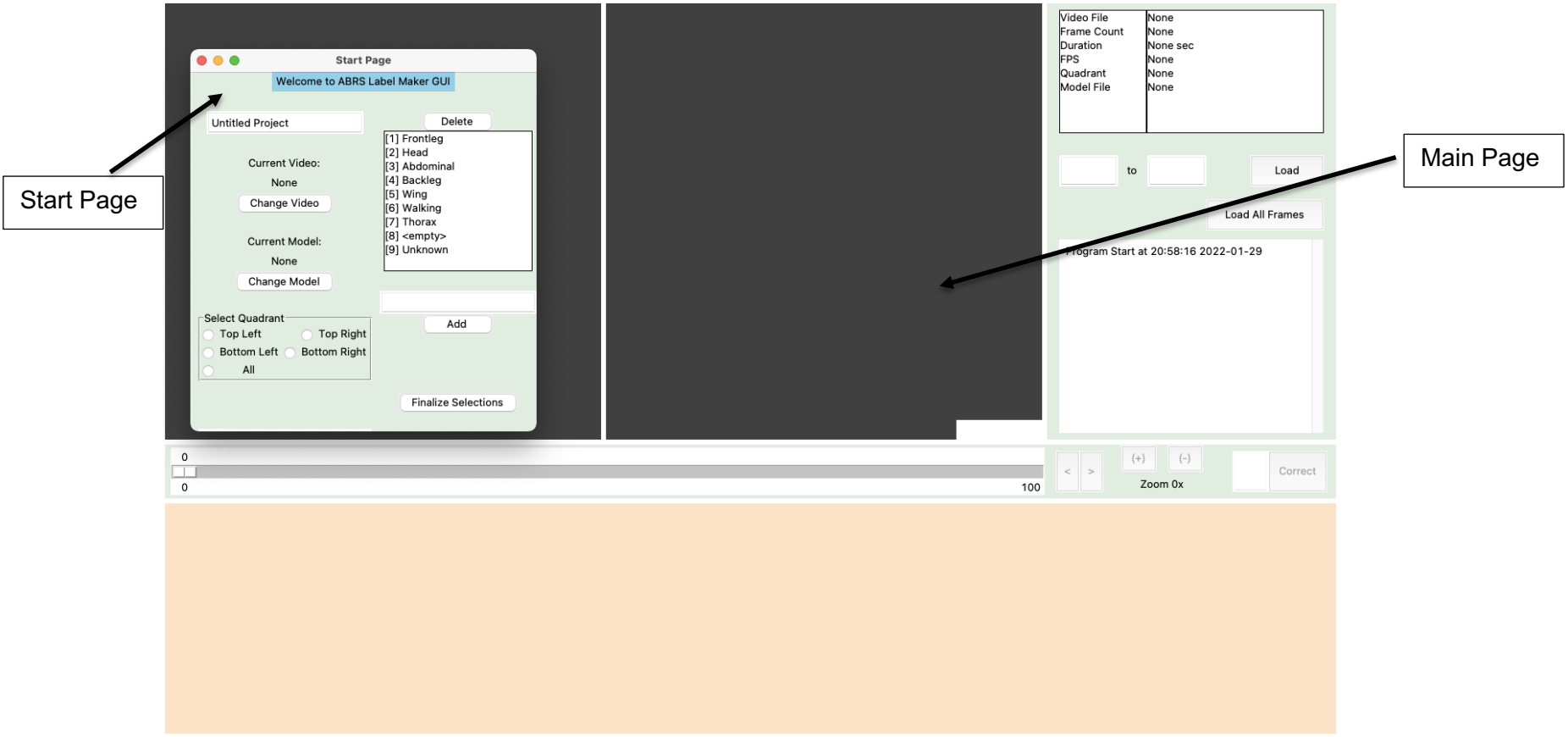
To run the GUI, highlight the cell/block holding the code by clicking anywhere on it. Once highlighted, run the code by pressing the play button (“Run selected cells and advance”) on the menu bar located at the top:


File Edit View Run Kernel Tabs Settings Help
/ ABRS /
Name
Last Modified
ABRS_modules.py
10 months ago
ABRSLabelMakerGUI.i...
12 minutes ago
Launcher
ABRSLabelMakerGUI.ipynb
Code
Python 3
[ ]:
##ABRS Label Maker GUI
Written by Primoz Kavbar
Edited by Venkata Manikanta Muriki
##################
START CODE
##################
import warnings
warnings.filterwarnings('ignore')
from tkinter import * # GUI toolkit
from tkinter import filedialog
from tkinter import messagebox
from tkinter import ttk
from tkinter import font
import matplotlib
import matplotlib.pyplot as plt
import math
import datetime
from matplotlib.backends.backend_tkagg import (
FigureCanvasTkAgg, NavigationToolbar2Tk)
# Implement the default Matplotlib key bindings.
from matplotlib.backend_bases import key_press_handler
from matplotlib.figure import Figure
import mplcursors
import cv2
import PIL.Image, PIL.ImageTk
from PIL import Image
import time
import numpy as np
This area is the file directory. Notice we are in the ABRS folder, and we see the ABRSLabelMakerGUI.pynb (Jupyter Python “py” Notebook “nb” file) and the ABRS modules .py file that has the required ABRS software.
Highlight the code block cell blue as shown by left clicking anywhere on the cell.
Pressing the Run button will execute the highlighted cell with code.
Blank circle means code is not running currently.
0 1 Python 3 | Idle
Saving completed
Mode: Edit
Ln 7, Col 19
ABRSLabelMakerGUI.ipynb

The first time running will take a bit longer than usual. You should see the top right Python3 circle filled in to signify start of the run:



Once the interface opens, you should see the Start Page pinned over the Main Page. The Start Page should be completed before the Main Page in the background can be accessed. This is how it would look the first time.

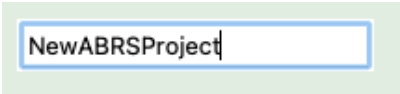


Start Page

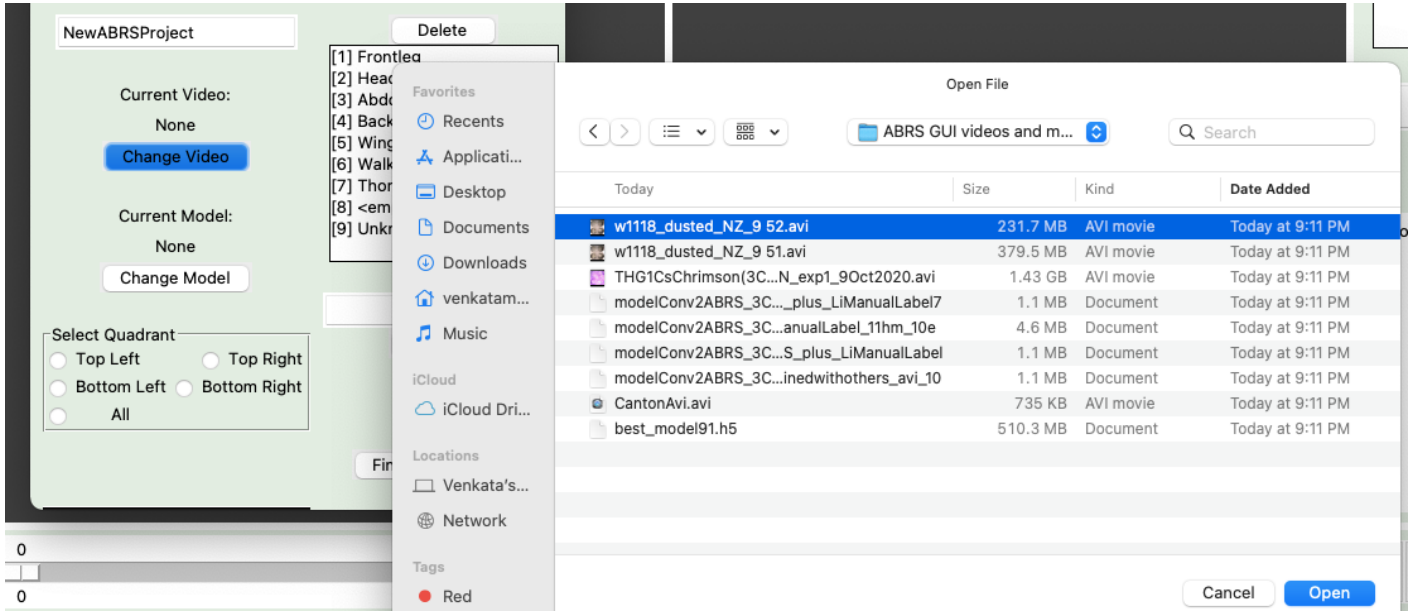
The Start Page is where you, the user, can feed in certain selections into the GUI before running it. “None” means nothing is selected for the options. These are the list of items available to customize.

On the left-hand side:

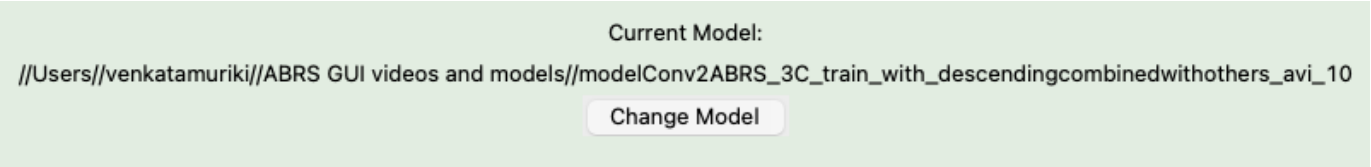
1. Choose a Project Name: This is where a project name should be typed in. A project name will be the name of the folder that the program will create and save the generated ethogram labels and ST images in. If the same project name is used for future runs, any other labels and images will be added to the same folder instead of overwriting previously saved content. The project name should be a valid folder name. To avoid unprecedented crashes in the program, it is recommended that you type in letters and/or numbers and/or spaces without any other special characters.



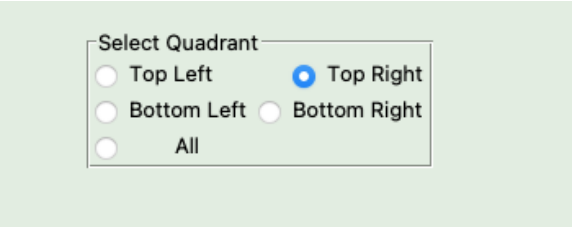
2. Choose the Current Video: Now click “Change Video” and browse the system for a valid .avi video file located anywhere on the system for the program to find and load the footage.



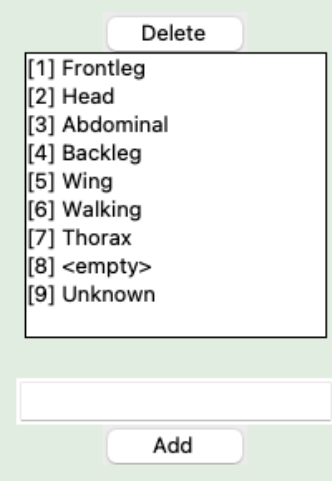
3. Choose the Current Model: Now click “Change Model” and browse the system for a valid model file located anywhere on the system for the program to find and analyze the video.



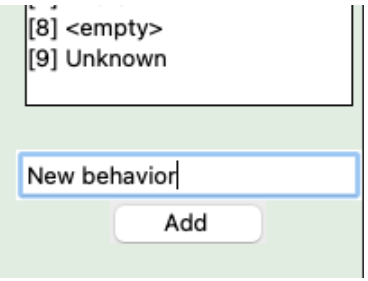
4. Selecting a quadrant: Research labs can have four feeds arranged in a 2 by 2 grid running on one frame to save data and time. This is where you can select which feed of the frame to load: Top Right, Top Left, Bottom Right, and Bottom Left. You can also analyze the entire frame if only one feed is available by selecting “All”, meaning all of the frame.



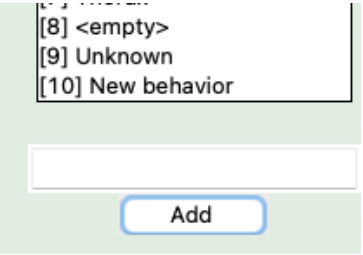
On the right-hand side, is a small window that shows the list of default labels that will be used by the GUI and those that will be displayed on the ethogram. The labels are numbered starting from 1. But you can easily change, add, or remove the list in the window.



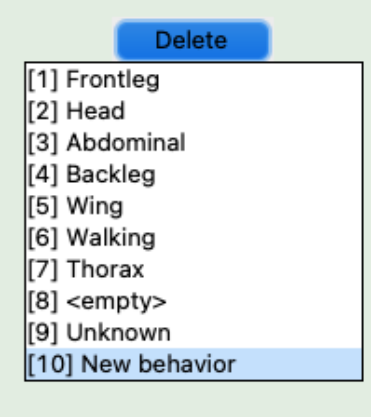
- Add: You can Add in certain labels after typing in the label name with proper syntax in the entry box.



The [label number] will be filled automatically so no need to type it in. There is no need to select the last label in the window. The entry will automatically be added as a new label to the end of the list.



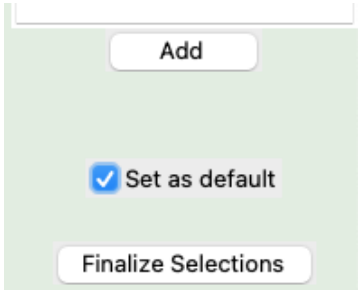
- Delete: You can select the label of interest and press Delete.
  - At least one label must be selected to delete.
  - Multiple labels in sequence can be selected by holding the right mouse button clicked and moving the cursor across the range of labels of interest and then all of these can be deleted after.



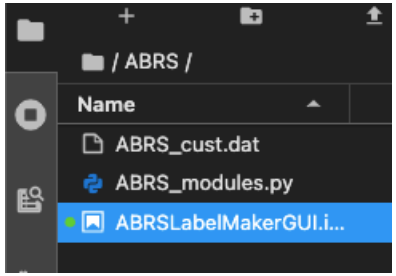
Double check whether the list of labels has the right label numbering to avoid confusion with correction.

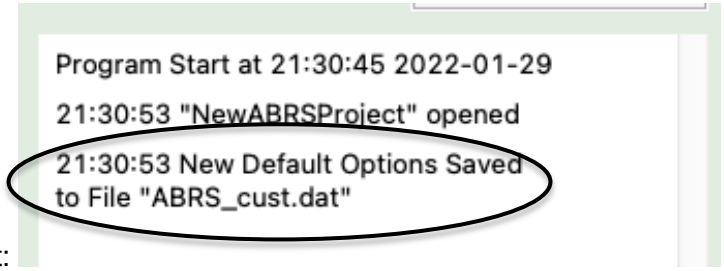
If it is the first time ever running the program, there will be no “Set as default” checkbox option available on the right-hand side of the Start Page, below the labels list window.

Before: 

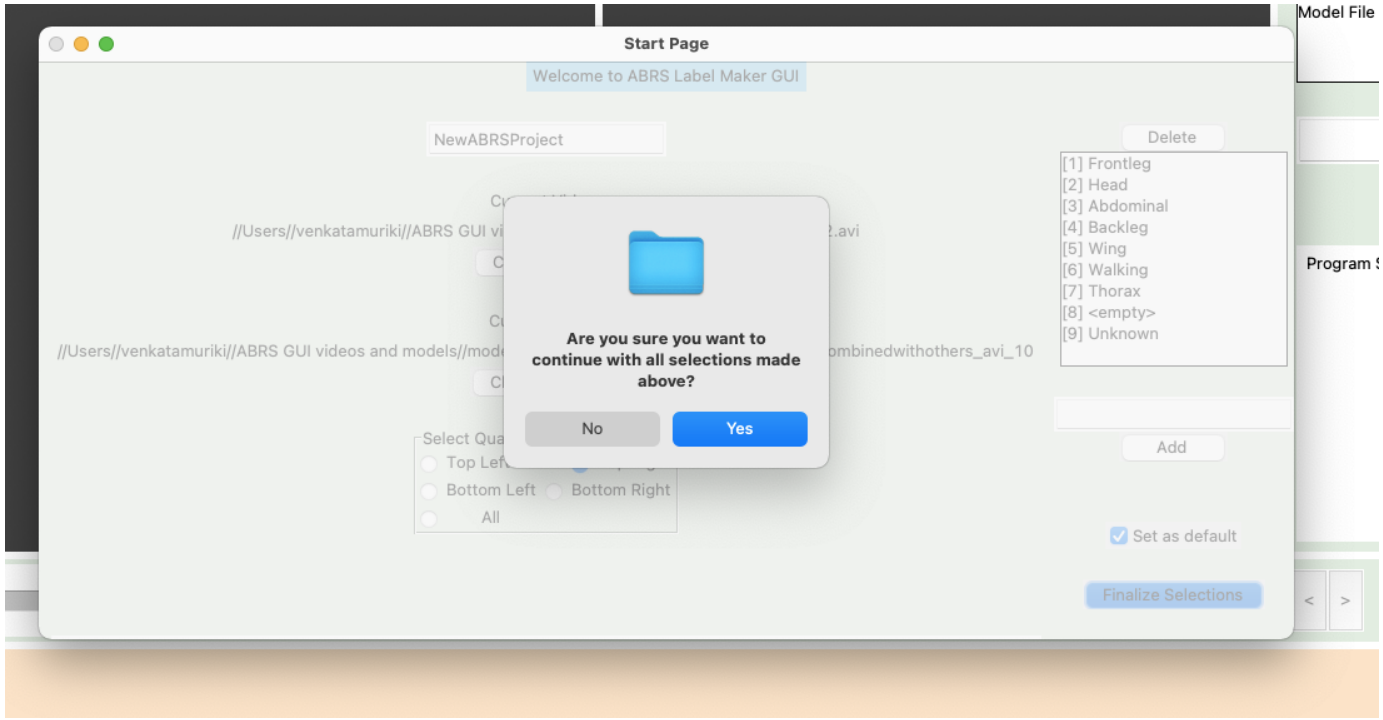
After: 

The valid selections made the first time will be set as default until you wish to change the default selections again during the second run and onward. These default selections will be stored in a file called “ABRS_cust.dat” in the current working directory.



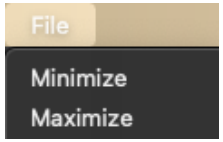
Ensure that this file gets added the first time you run the program. Every time the default options are changed, the History window should print out: 

Click “Finalize selections” if you would like to move forward with the selections made. The program will prompt you for a confirmation.



Layout of the Main Page

Now you are on the Main Page. You should see that in the History window, that the “Program Start at <time and date stamp>” line should be output. Next, the “<timestamp> <Project Name> opened” line should print. See below for the full layout of the GUI explained. The GUI should be used in full-screen the entire time. You should minimize and maximize the GUI by going to File → and

pressing 

as desired. The GUI was developed using a Python software that is fixed to work on certain resolutions. If the window is resized or manipulated, the layout of the GUI will be disturbed, the program may lag, and functional issues of the buttons could occur.

You should also see that the entire bottom half and the two, frames-displaying windows (in black) should be inactive because no frames are loaded so the ethogram, horizontal scrollbar, correcting labels, and zooming in should be disabled.

See the next two pages for the layout of the GUI before and after loading frames.

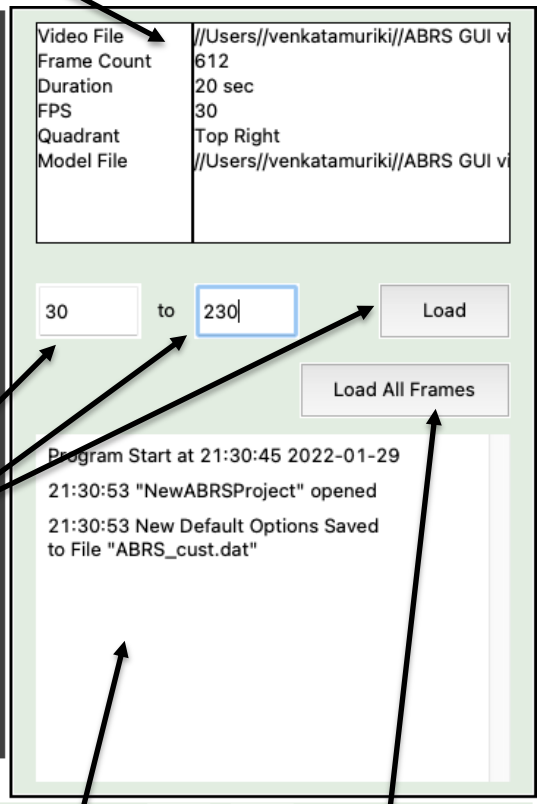
Before Frames Loaded:

Information box that is horizontally scrollable to view the current video and its details, quadrant loaded, and model loaded.

Currently empty but will be populated by the raw video freeze-frame of a certain current timepoint in the video.

Will be populated by the ST image version of the freeze-frame at the same time point as the raw frame.

The start frame and end frame entries and the respective load button.



History box that prints important commands executed, which the user can reference.

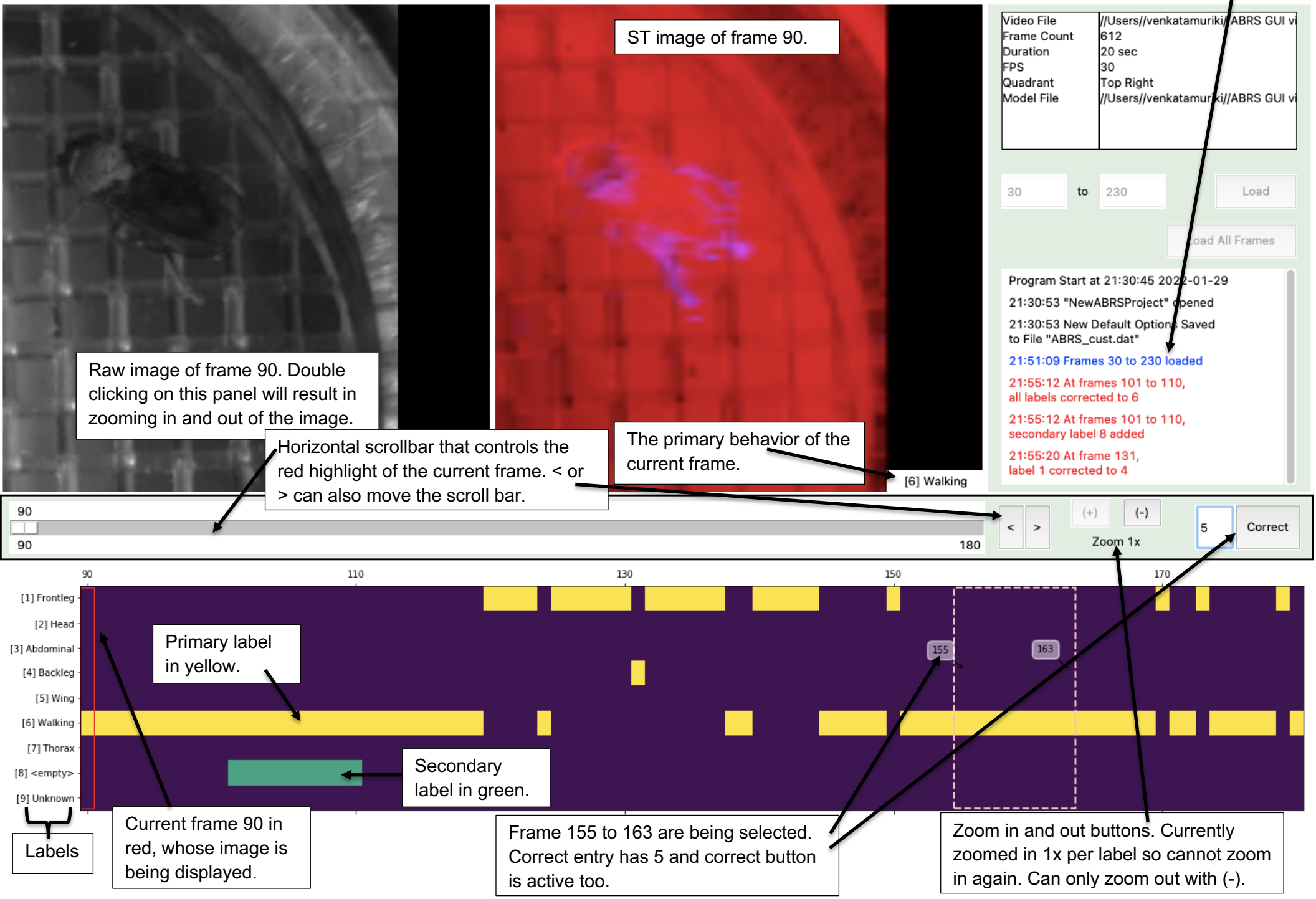
Loads all frames, irrespective, from 0 to “frame count”, doesn’t account for frame entries above.

Will be populated by the ethogram generated by the ABRS assigning behaviors to the subject.

After Frames Loaded:

Notice that frames 30-230 were loaded, but the horizontal scrollbar shows 90-180, which are the frames being zoomed in on the ethogram.

Notice that the History box is populated with the blue command describing what frames were loaded and red commands describing what and where corrections were made.



Loading frames

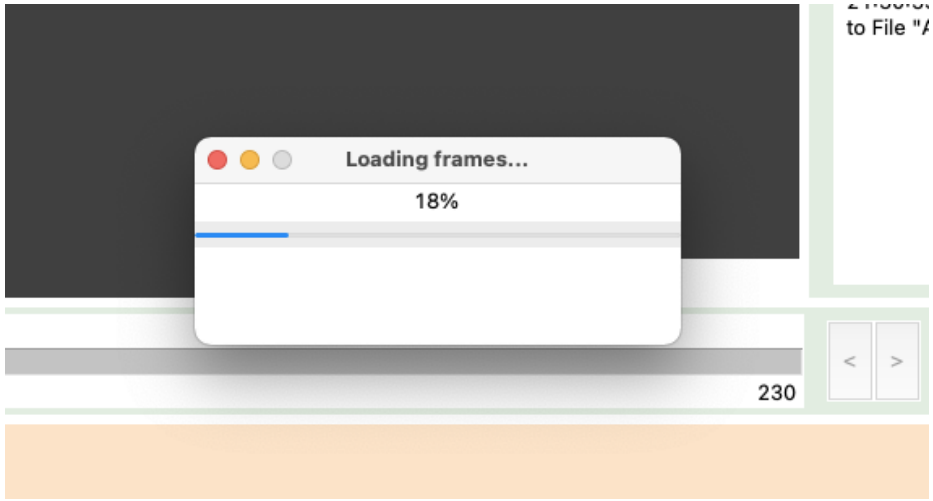
Now that you have been introduced to the layout of the GUI, let's talk about loading frames (freeze-frame). The frame captures an instant of the video for the model to analyze.

I. You can load frames by number. Type in the start frame in the left entry box and the end frame in the right entry box. Frames start from 0 because a video starts from time point 0. If you want to load the first 200 frames of the video, you can type "0" to "199". You can then press the Load button.

II. You can also Load All Frames. This will essentially load the entire video and it will take a while, so the GUI will prompt you for a confirmation.

The start and end frame entries should be valid whole numbers greater than equal to 0. The start frame should also be less than the end frame. The last frame in the video cannot be greater than but can be equal to the frame count number specified in the top right corner.

A progress bar should open that shows the progress of loading the desired frames along with a percentage number. If the progress bar does not show up or freezes for a while, then there is an error with reading the video, so quit the GUI and find the error message located at the bottom of the code block, highlighted in red.



Navigating through the video frames

The inactive, empty spaces in the Main Page should now be populated with the raw frame image on the left black window, ST image of the frame on the right black window, and the ethogram in the bottom panel. The History window should update you with what frame numbers were loaded in blue. Usually, the first few dozen frames are not as accurately labeled by the behavior because the model needs ~30-60 to get a large enough temporal context, so you might not see the subject (example: fly) immediately in the video images. You should also notice that the load buttons and frame entries are disabled, since multiple loads cannot be done in a single run.

Navigating the video frames can be done via the horizontal scrollbar just below the images. You can also use the 

buttons located to the right of the scrollbar. Notice the red rectangle on the ethogram that highlights the frame being displayed is synced with the scrollbar.

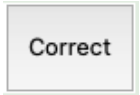
You can also double click on the raw frame image on the left to zoom into the image and you can double click again to zoom out. You cannot zoom in/out of the ST image on the right, it will always stay zoomed in.

Correcting the primary label of frames

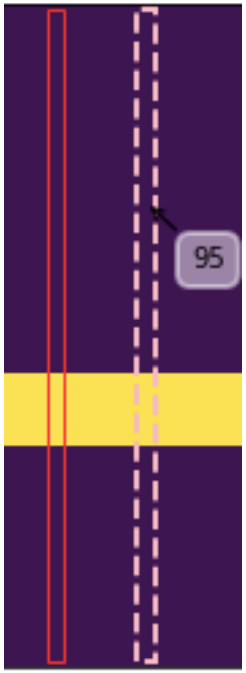
- To find ONE frame whose primary label must be corrected, use the frame number tick marks on the ethogram or the red rectangle connected to the horizontal scrollbar that highlights the current frame. [Helpful tip: Matching the label names on the y-axis to the yellow label the frame was given on the ethogram might be complicated, but the red-highlighted frame's ST image panel that has the assigned label written in the bottom right corner.] By left-clicking with a mouse on that part of the ethogram selects the frame with a white dashed rectangle. A mini pop-up on the ethogram near the click site should confirm the frame that you selected. The 

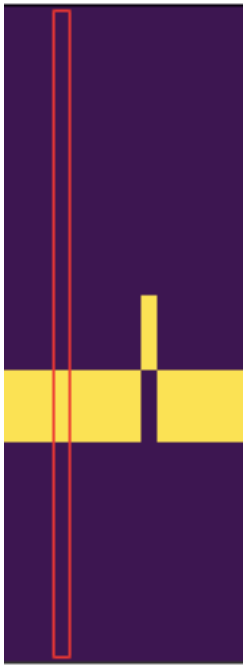
or the 

button and its entry should now be enabled for you to either zoom in (if not already zoomed in) or to type in a label number for correction, respectively.

- Now to correct the primary label, enter the correct label number in the entry box located on the right side of the middle panel, to the right of the Zoom buttons. Make sure that the label number is in the range of labels specified, because otherwise the system will throw a warning until valid syntax is followed.
- Now hit 

, and you should immediately see your changes realized on the ethogram.
- There should be the white dashed rectangle that lingers after the correction is made to show you the frame where you just corrected. This selection MUST be cleared by left clicking anywhere on the ethogram before new corrections or even selections can be made. If this sequence of events is now followed, then the GUI will freeze and run into an error.
- The correct and zoom elements should be disabled once again until a new selection is made.



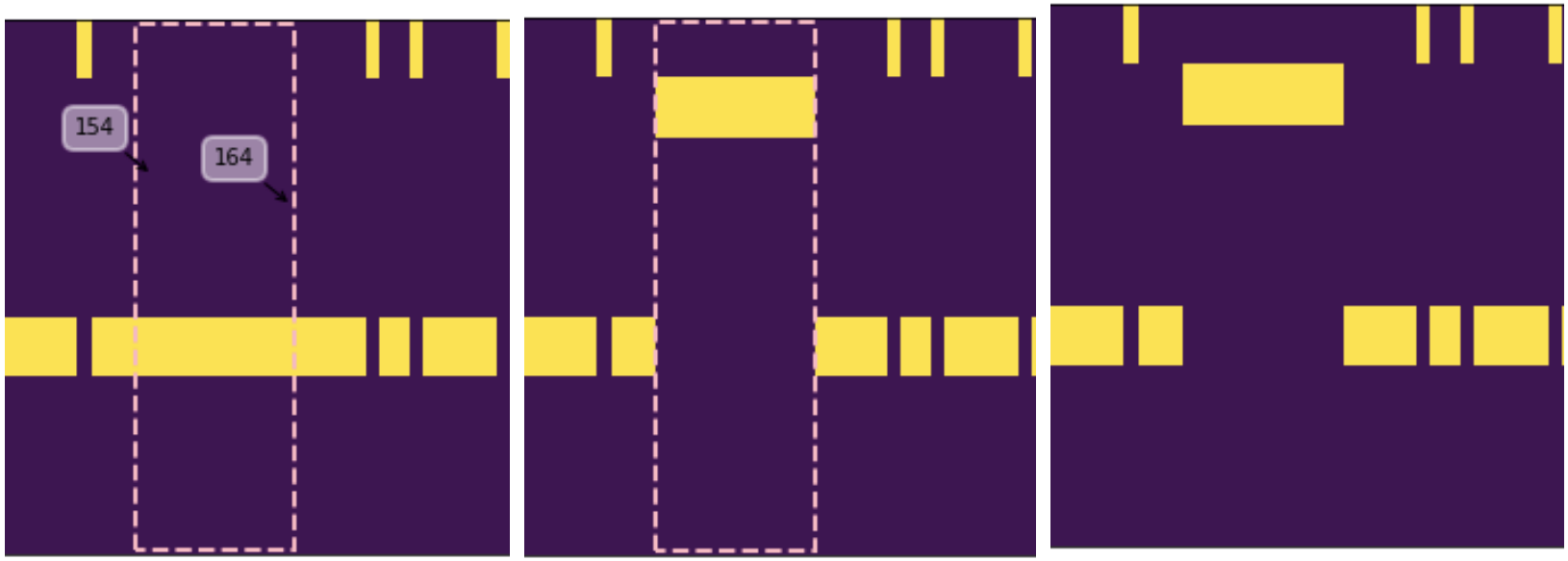



- To select a range of sequential frames whose labels must be corrected, you must left click twice on the ethogram. This can be done two ways. You can either click the first frame to select first and then the last frame to select with the second click or

vice versa. All the frames including and between the two clicked frames are selected by the white, dashed rectangle. Each click shows you which frame was selected with two separate mini pop-ups. Again, using the red rectangle to guide where to click to select the desired frame can be very helpful. After making the first click, the zoom and correct entry and buttons will be enabled until a correction is made, and once again they will be disabled until a new selection can be made.

- Now to correct the primary label of these selected frames, type the new label in the entry box and click 

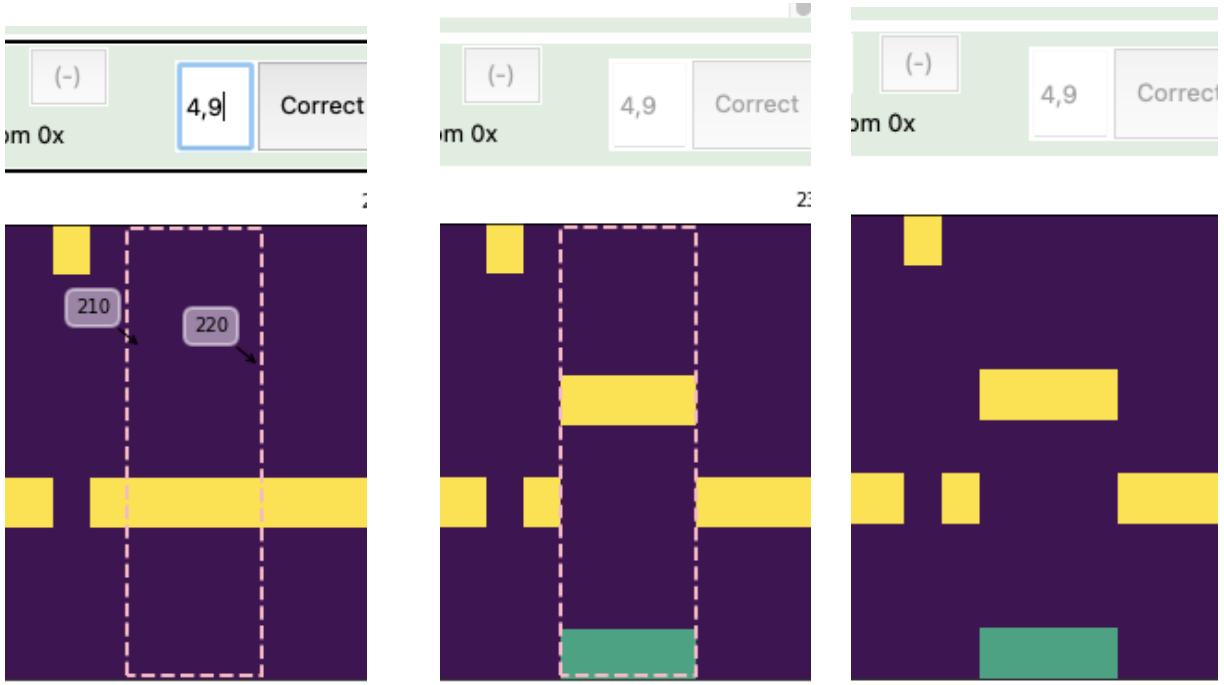
. Your changes should be reflected on the ethogram and again, you should clear the white selection that lingers before you can make new selections/corrections.



Correcting the primary label of frames and adding a secondary label

This section describes correcting the primary (yellow) label as you did above, but also being able to add a secondary (green) label to the same frame or frames. The researcher can use this function to their liking, but one use for adding a secondary label is for the researcher to mark the frame with another behavior in addition to a main one.

Again, selection for one or a multiple range of frames can be done similarly as described above. However, unlike before, in the entry box you should type in “[new primary label], [new secondary label]” and press Correct. Notice the comma that separates the two label numbers. The order is important here because entering “6,2” and correcting a frame that is labeled “[5] Wing Cleaning” will change the yellow label from 5 to 6 and add a green label at 2 on the ethogram. The exact syntax described above and with the numbers in the proper range should be entered, or the system will throw a warning. You can even change previous secondary labels by selecting the desired frame(s) to change and specifying the same primary label but a new secondary label. Again, it is important to clear the white dashed selection before new selections or corrections can be made.

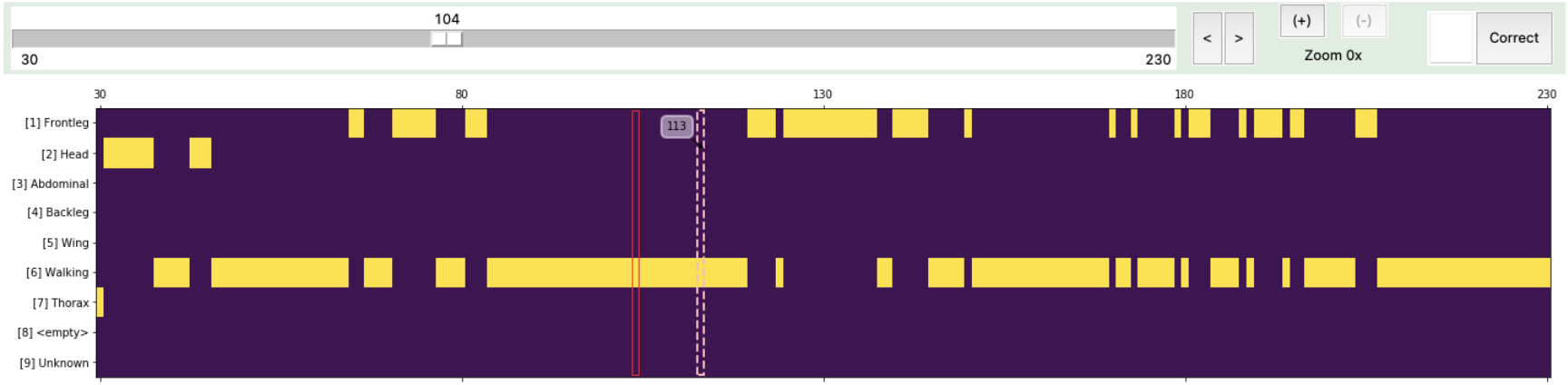


Zooming in/out on ethogram

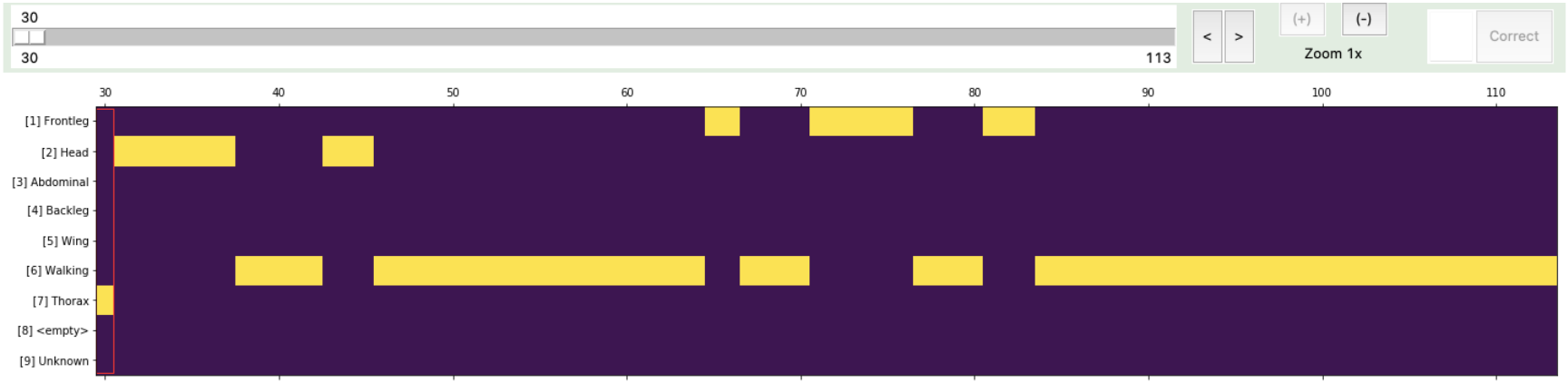
Zooming is only available if you are not already zoomed in and if the user has made a selection. There are **three** ways to zoom in:

1. If you left click once on the left half of the ethogram and select one frame, clicking the (+) button will zoom in on the ethogram from the first loaded frame to the clicked frame (Because it might be tough to exactly click the first frame on the edge of the ethogram if deciding to go with way #3. Same reason for #2, end frame is on the other edge.)

Before:

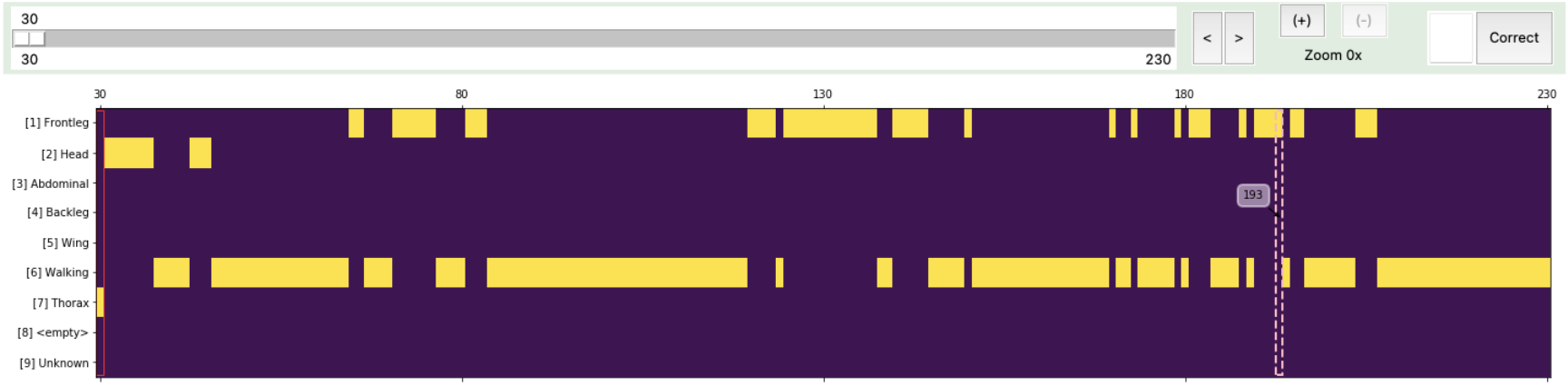


After:

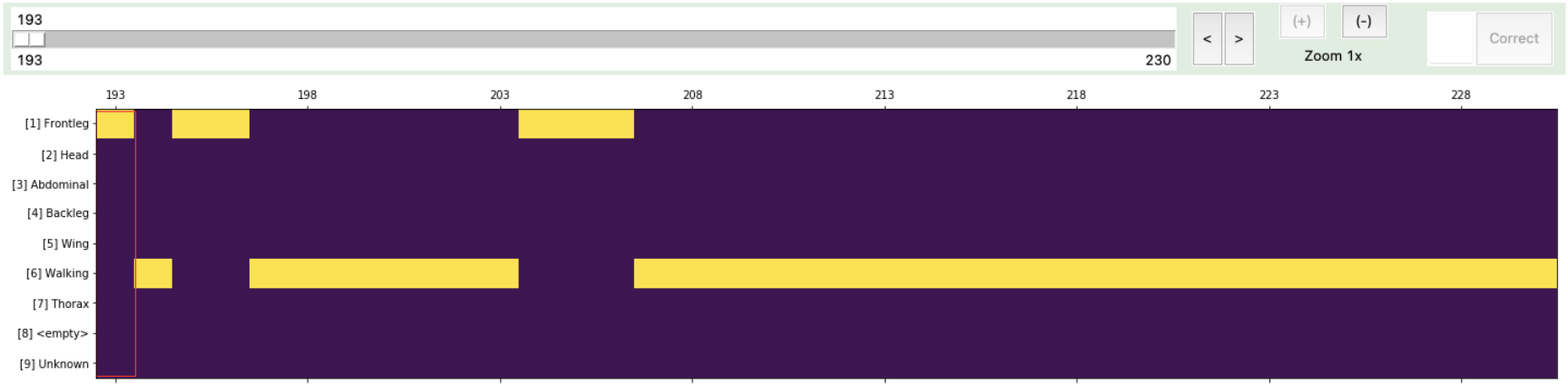


2. If you left click once on the right half of the ethogram and select one frame, clicking the (+) button will zoom in on the ethogram from the clicked frame to the last loaded frame

Before:

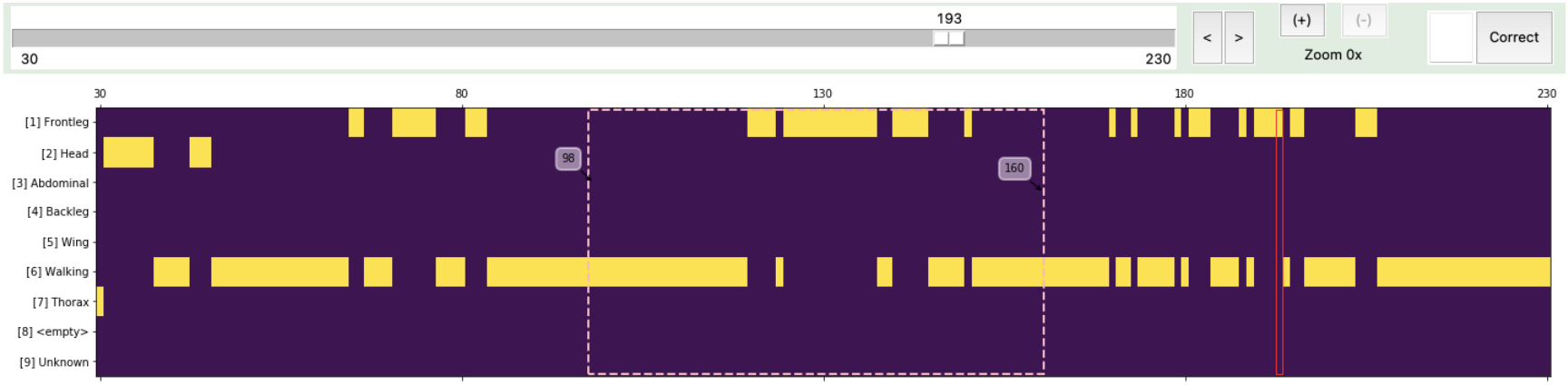


After:



3. If you click twice and select multiple frames and press the (+) button to zoom in on that range of frames. As described earlier, you can select backwards too and still, zooming will be done from the first chronological frame to the last.

Before:



After:

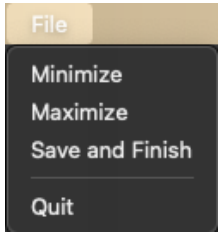
The zoom indicator should change from Zoom 0x to Zoom 1x to let the user know when they are zoomed in.

Zooming OUT:
You can zoom out any time after zooming in, regardless of any selection being made.

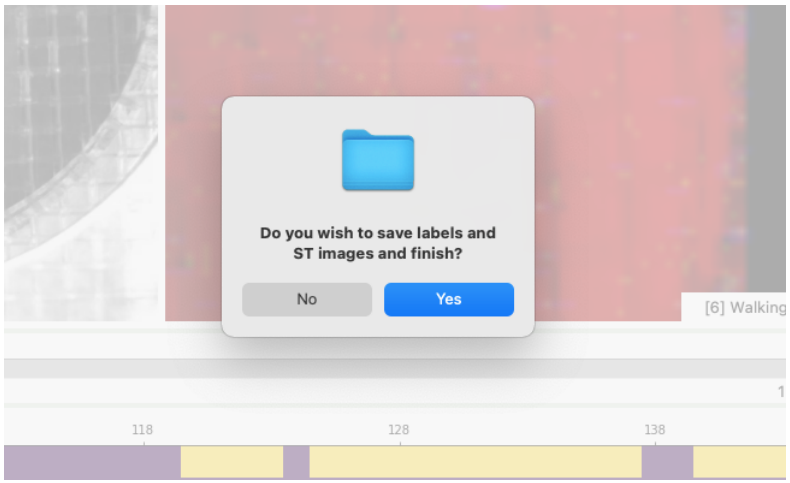
You can zoom out without any selections any time by just clicking (-).

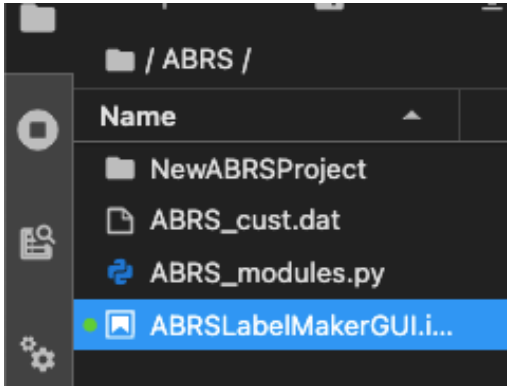
Quitting/Saving and Finishing

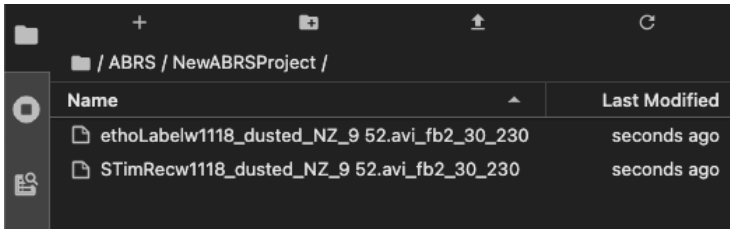
To quit the program without saving any of your progress, go to File → Quit. Since the application is in full-screen mode, the menu bar might be hidden on some OS. But when moving the cursor to the top, it should drop down to show the file options.



To save the ethogram labels or ST images for the loaded frames, go to File → Save and Finish. The labels and ST images should be labeled by the frames numbers loaded in a folder named after the project name:







In the event of a crash of the program

It is possible that GUI can run into new errors as new types of videos are downloaded and new models are being used. You can tell when the program crashed if the buttons are frozen or if you minimize out of the application and scroll to the bottom of the code and see lines of errors or warnings in red. In this case, quit out of the program or even quit out of the python itself and please report the error to the creators.

Contact

Contact creator: Primoz Ravbar at primoz.ravbar@gmail.com
Contact editor: Venk Muriki at vmuriki@ucsb.edu